Modelling Facial Communication Between an Animator and a Synthetic Actor in Real Time

Nadia Magnenat Thalmann[*] [**]
Antoine Cazedevals [**] and Daniel Thalmann[***] [**]

[*] MIRALab, CUI, University of Geneva
24 rue du Général-Dufour, CH 1205 Geneva, Switzerland
Email: thalmann@uni2a.unige.ch

[**] MIRALab, HEC, University of Montreal, Canada

[***] Computer Graphics Lab
Swiss Federal Institute of Technology
CH 1015 Lausanne, Switzerland
Email: thalmann@eldi.epfl.ch

Abstract

This paper describes methods for acquiring and analyzing in real-time the motion of human faces. It proposes a model based on the use of snakes and image processing techniques. It explains how to generate real-time facial animation corresponding to the recorded motion. It also proposed a strategy for the communication between animators and synthetic actors.

Keywords: facial analysis, facial animation, snake, muscles

1. Introduction

There has been extensive research done on basic facial animation and several models have been proposed. Early methods proposed by Parke (1975, 1982) are based on ad-hoc parameterized models. Platt and Badler (1981) introduced facial expressions manipulated by applying forces to the elastically connected skin mesh via underlying simplified muscles. Their system is based on the Facial Action Coding System (FACS) developed by Ekman and Friesen (1975). Waters (1987) developed a face model which includes two types of muscles: linear/parallel muscles that pull, and sphincter muscles that squeeze. Nahas et al. (1988) proposed a method based on B-splines; motion of the face is obtained by moving the control points. Magnenat-Thalmann et al. (1988) provided another approach to simulate a muscle action by using a procedure called an Abstract Muscle Action (AMA) procedure.

Terzopoulos and Waters (1990) proposed a physics-based model three layered deformable lattice structures for facial tissues: skin, subcutaneous fatty tissue, and muscles. Parke (1991) reviews different parameterization mechanism used in different previously proposed models and introduces the future guidelines for ideal control parameterization and interface. Kalra et al. (1991) introduced a multi-layer approach where, at each level, the degree of abstraction increases. They also described another approach to deform the facial skin surface using rational free form deformations (Kalra et al. 1992). DiPaola (1991) proposed a facial animation system allowing the extension of the range of facial types. We

may also mention efforts for lip synchronization and speech automation by several authors (Lewis and Parke; 1987; Hill et al. 1988; Magnenat-Thalmann et al. 1987; Lewis 1991) .

Recently several authors have proposed new facial animation techniques which are based on the information derived from human performances. The information extracted is used for controlling the facial animation. These performance driven techniques provide a very realistic rendering and motion of the face. Williams (1990) used a texture map based technique with points on the surface of the real face. Mase and Pentland (1990) apply optical flow and principal direction analysis for lip reading. Terzopoulos and Waters (1991) reported on techniques for estimating face muscle contraction parameters from video sequences. Kurihara and Arai (1991) introduced a new transformation method for modeling and animating the face using photographs of an individual face. Waters and Terzopoulos (1991) modeled and animated faces using scanned data obtained from a radial laser scanner. Saji et al. (1992) introduced a new method called "Lighting Switch Photometry" to extract 3D shapes from the moving human face. Kato et al (1992) use isodensity maps for the description and the synthesis of facial expressions. These techniques do not process the information extraction in real-time. However, real-time facial animation driven by an interactive input device was reported by DeGraf (1989).

This paper describes a model for real-time analysis and synthesis of facial expression and emotion recognition. Section 2 explains the basic principles of the real-time analysis based on snakes and image processing techniques. Section 3 describes the way of recognizing simple expressions and emotions. The synthesis of facial animation from information extracted during the analysis phase is developed in Section 4. Section 5 proposed the use of our approach for the communication between an animator and synthetic actors. Finally implementation issues are discussed in Section 6.

2. The analysis method

2.1 The use of snakes

Our recognition method is based on snakes as introduced by Terzopoulos and Waters (1991). A snake is a dynamic deformable 2D contour in the x-y plane. A discrete snake is a set of nodes with time varying positions. The nodes are coupled by internal forces making the snake acting like a series of springs resisting compression and a thin wire resisting bending. To create an interactive discrete snake, nodal masses are set to zero and the expression forces are introduced into the equations of motion for dynamic node/spring system. The resulting equation for a node i (1 = 1,...,N) is as follows

$$\gamma_i \frac{dx_i}{dt} + \alpha_i + \beta_i = f_i$$

where γ_i is a velocity-dependent damping constant, α_i are forces resisiting compression, β_i are forces resisting bending and f_i are external forces. To turn the deformable contour into a discrete snake, Terzopoulos and Waters make it responsive to a force field derived from the image. They express the force field which influences the snake's shape and motion through a time-varying potential function. To compute the potential, they apply a discrete smoothing filter consisting of 4-neighbor local averaging of the pixel intensities allowed by the application of a discrete approximation.

Our approach is different from Terzopoulos-Waters approach because we need to analyze the emotion in real-time. Instead of using a filter which globally transforms the image into a planar force field, we apply the filter in the neighborhood of the nodes of the snake. We only use a snake for the mouth; the rest of the information (jaw, eyebrows, eyes) is obtained by fast image-processing techniques.

For the mouth snake, we use the method illustrated by Fig.1. On this figure, we may see the snake working around the mouth. The small lines starting from the nodes show the direction of the forces generated. Because of the elasticity of the snake, if there is no force, the snake tends to contract and becomes a single point.

Fig.1 Visualization of the recognition system

To generate the forces, we extract a 6x6 matrix M_1 around the node P as shown in Fig.2. The node's coordinates are converted into integer values.

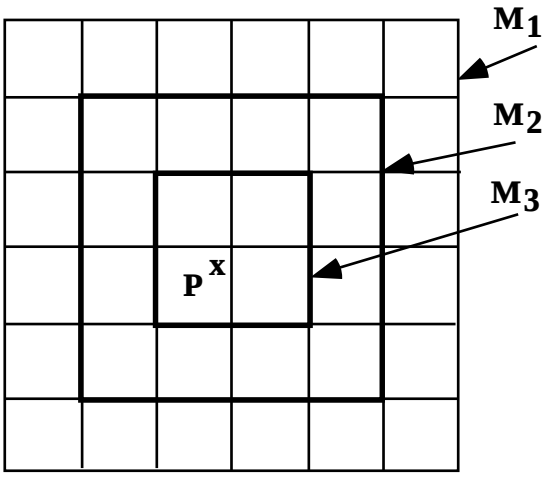

Fig.2. Computation of the filter

To compute the potential, we first transform the RGB information into intensity information; for example, red contributes 30%, green 59% and blue 11%. Secondly, we apply on the matrix a discrete smoothing filter followed by a discrete approximation to the gradient operator. The first filter is applied on the 4x4 square matrix M_2 included in M1. The second filter is applied on the 2x2 square matrix M_3 at the center of the matrix. With these 4 points, we can interpolate bilinearly the force vector.

The snake we used has the same tension constant for all the springs. The rigidity constant is less at the corners of the mouth.

Because of real-time constraints, the snake can have some difficulties following the mouth's edges with good accuracy. The main problem is guiding the snake on the x-axis because the mouth has no strong vertical edge.

In our system, the snake is forced to stay at the center of the head. On Fig.1, we may see two circles near the edges of the neck. These two circles have the same y positions as the endpoints of the snake.

As shown in Fig.3, to determine the x position for the left circle (abscissa of T), we start at the left of the image (S) and scan to the right until an edge is detected. To detect the edge, we first calculate an average of the intensity of the right point and its 4 neighbors:

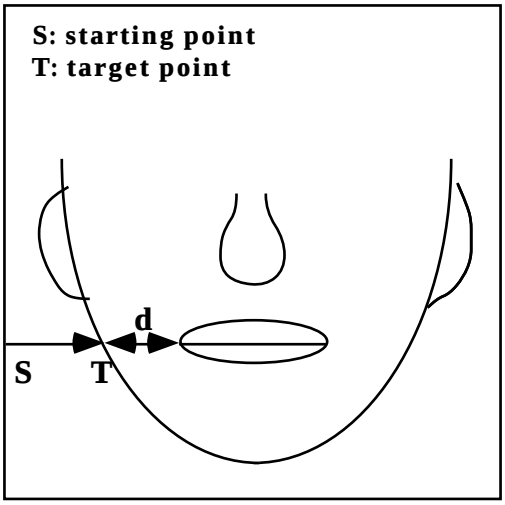

Fig.3. Computation of distance d

Secondly, we compare the result with the point where we are, and if there is not an important difference of intensity, we go right of 1 point.

We then use the information provided by the snake to find the distance d.

We perform the same processing for the right side. We then compare the distance d with the same distance d_0 stored during the initial procedure described later on. If d is less than d_0, we apply some forces to the extremum of the snake and reduce the difference. The magnitude of the generated force is computed as follows:

$$F = k \, | d - d_0 |^3$$

2.2. Image processing methods

For the **jaw**, we consider that the lower part of the lower lip (using information given by the snake) is moving with the jaw i.e. if the law opens, the lower part goes down with the jaw. On Fig.1, we may see a circle which uses its y coordinate to find the position of the jaw.

For the **nose**, we use the center point of the upper part of the mouth (also using the snake) and we scan upwards until an edge is detected. As we assume that the illumination is very strong, the edge should belong to the shadow of the nose. The circle on the nose on Fig.1 indicates the position found by the program.

For the **eyebrows**, we use the same principles as the nose. We start from the forehead and scan downwards until we detect an edge. This should be the eyebrow. On Fig.1, we show two circles on the forehead (the starting points) and under them, the two circles indicating the positions for the eyebrows.

For the **eyes**, we define a rectangular region around the eyes (using the position of the nose and eyebrows) and we count the number of white points in the region. If the number of white points is under a threshold value, we consider the eye as closed. On Fig. 1, we show rectangles defining the region. In order to process the analysis in real-time, we only consider one point out of two.

A first step is always necessary to store information when the person has a neutral expression. To determine the intensity of an expression, we compare the information of the current frame with the corresponding information for the neutral expression. For example, if the distance between one eyebrow and the nose is larger than the initial distance, we know that this eyebrow has a higher position. A more general methodology for expression and emotion recognition is explained in the next section.

3. Emotion recognition

In order to recognize expressions and emotions for a new person, the system first needs to know the values of parameters corresponding to the neutral expression. This reference data capture is performed using the snake technique; the person should keep his/her face very quiet and then push the INIT button on the control panel of the program to record these reference parameters. Typical reference parameters are:

the reference left eyebrow height: le_0

the reference right eyebrow height: re_0

the reference mouth width: mw_0

the reference mouth height: mh_0

Fig.4 shows the extracted information.

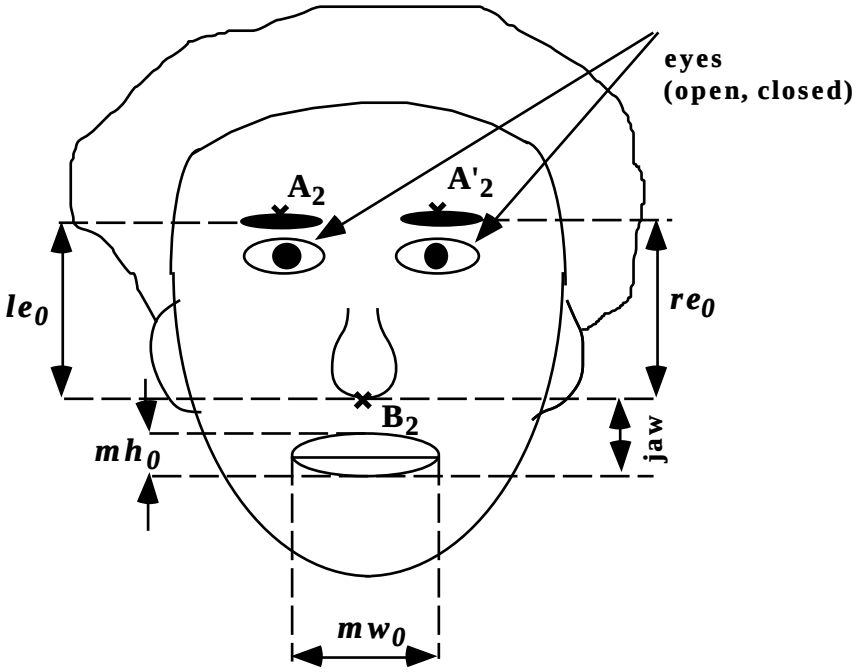

Fig.4. The extracted information

The expression/emotion recognizer compares values obtained in real-time to the reference values to decide the type of expression/emotion. The rules are generally too simple but they may be easily changed. For example, we give six rules using the following notations for the current parameter values:

the reference left eyebrow height: le
the reference right eyebrow height: re
the reference mouth width: mw
the reference mouth height: mh

examples of rules:

1) open mouth:
if $mw/mh < 2$ then "mouth in O"

2) lowered eyebrows:
if $le < 0.9*le_0$ and $re < 0.9*re_0$ then "not happy"

3) open mouth but especially horizontally
if $mw*mh > 2*mw_0*mh_0$ and $2 < mw/mh < 2.5$ then "laugh"

4) open mouth but not circular and eyebrows lightly raised:
if $mw*mh > 2*mw_0*mh_0$ and $mw/mh < 2$ and $0.9*le_0 < le < 1.1*le_0$ and $0.9*re_0 < re < 1.1*re_0$ then "strong laugh"

5) mouth in O and high eyebrows:
if $mw*mh > 2*mw_0*mh_0$ and $mw/mh < 1.8$ and $le > 1.1*le_0$ and $re > 1.1*re_0$ then "afraid"

6) half open mouth and high eyebrows:
if $mw*mh > 1.5*mw_0*mh_0$ and $mw*mh < 2*mw_0*mh_0$ and $mw/mh < 1.8$ and $le > 1.1*le_0$ and $re > 1.1*re_0$ then "surprised"

4. Synthesis of facial animation

One of the first application of the face analysis is the generation of the same expressions on a synthetic actor (see Fig. 5). For this purpose, we work at the low level of minimal perceptible actions as defined in our multilayered Facial Animation system SMILE (Kalra et al. 1991). Only 8 minimal perceptible actions are used:

open_jaw
raise_left_eye, raise_right_eye
close_left_eyelid, close_right_eyelid
raise_upper_lip, raise_lower_lip
pull_mid_lips

This is enough to control the main changes of the face, but not to copy exactly the expression of the real face, but again, our purpose is real-time processing more than accuracy. The facial deformations are performed using Rational Free-Form Deformations as defined by Kalra et al. (1992).

Fig.5. Example of real time recognition/synthesis process

5. Communication actor-animator

Based on the model described in this paper, we are developing an animator-actor communication as described in Magnenat Thalmann- Thalmann (1991).

As shown in Fig. 6, the system is mainly an inference system with facial and gesture data as input channels and face and hand animation sequences as output channels. The input data is captured in two ways: the methodology described previously for face expressions and datagloves for hand motions.

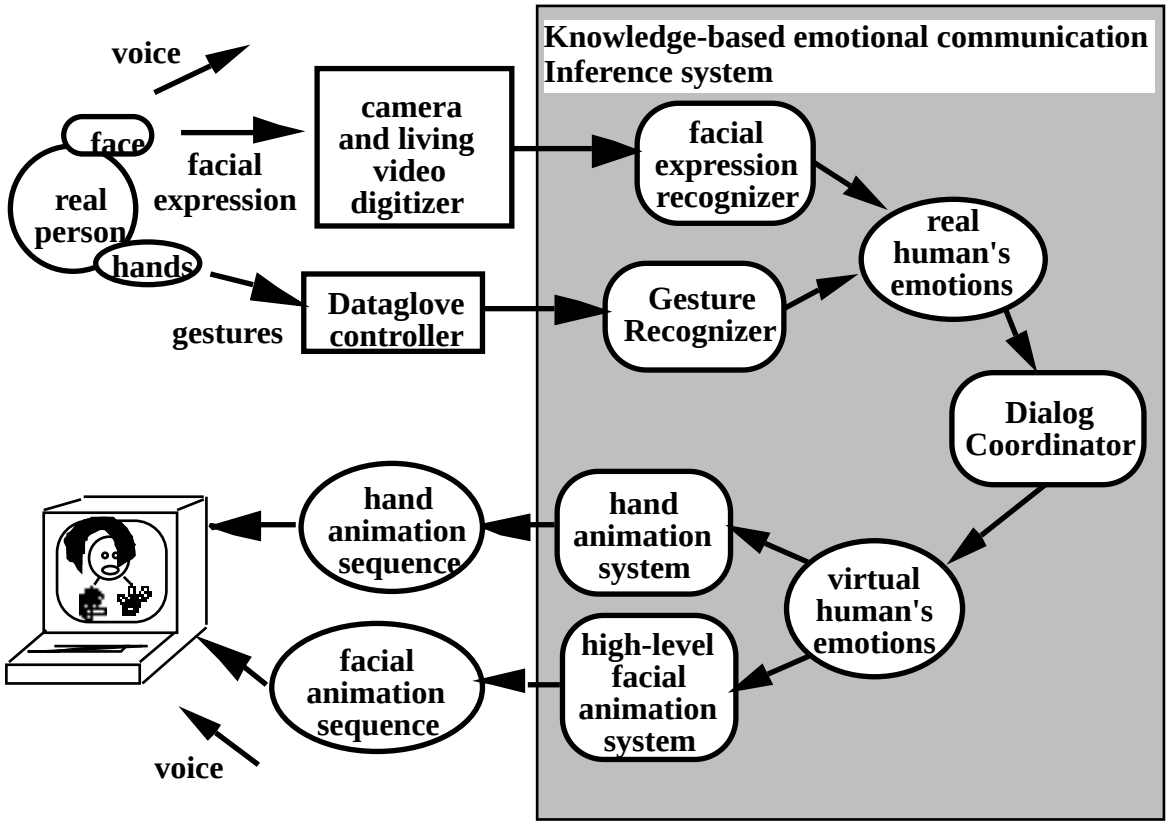

Fig.6 Organization of the proposed system

The development of the inference system is divided into three subsystems:

i) a subsystem for the recognition of emotions from facial expressions, head-and-shoulder gestures, hand gestures and possibly speech

ii) a subsystem for the synthesis of facial expressions and hand motions for a given emotion and speech

iii) a subsystem for the dialog coordination between input and output emotions

This last subsystem is a rule-based system: it should decide how the virtual actor will behave based on the behavior of the real human. The dialog coordinator analyzes the humor and behavior of the user based on the facial expressions and gestures. It then decides which emotions (sequences of expressions) and gestures (sequence of postures) should be generated by the animation system. For the design of correspondence rules, our approach is based on existing work in applied psychology, in particular in the area of non-verbal communication.

6. Implementation

The recognition program is working on any SG IRIS workstation, but real-time (about 10 frames/sec) is obtained on the 4D/440 VGX for simple emotions as discribed here. The video input is obtained using a professional camera connected to the SG Living Video Digitizer. The program has been developed in C using the Fifth Dimension object-oriented toolkit (Turner et al. 1990). The program interface only uses buttons and panels.

The SMILE system and the recognition program work on different UNIX processes. As the animation of the virtual face should be done at the same time as the recognition, both programs should exchange information. We use the UNIX inter-process communication protocol (ipc) which allows processes to send and receive messages.

7. Conclusion

This paper has shown that recognition of emotions in real-time is possible. This recognition may be used for generation of expressions by synthetic actors with same expressions or for truly communication between animators and synthetic actors. The main weakness of our program is that it uses techniques which require information from the previous frame to analyze the next one. This means that if the program does not succeed in extracting information from one frame, it cannot later extract information from the next ones. Only a tool analyzing frames independently could solve this problem, but in this case, real-time processing could not be possible any longer.

Acknowledgment

The authors are grateful to Prem Kalra for his help with the facial animation system. The project was sponsored by the Fonds National Suisse pour la Recherche Scientifique and the Natural Sciences and Engineering Council of Canada.

References

deGraf B (1989) in State of the Art in Facial Animation, SIGGRAPH '89 Course Notes No. 26, pp. 10-20.

DiPaola S (1991) Extending the Range of Facial Types, Journal of Visualization and Computer Animation, Vol.2, No4, pp.129-131.

Ekman P, Friesen WV (1975), Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues, Printice-Hall

Hill DR, Pearce A, Wyvill B (1988), Animating Speech: An Automated Approach Using Speech Synthesised by Rules, The Visual Computer, Vol. 3, No. 5, pp. 277-289.

Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1991) SMILE : A Multilayered Facial Animation System, Proc IFIP WG 5.10, Tokyo, Japan (Ed Kunii Tosiyasu L) pp. 189-198.

Kalra P, Mangili A, Magnenat Thalmann N, Thalmann D (1992) Simulation of Facial Muscle Actions Based on Rational Free Form Deformations, Proc. Eurographics '92, pp.59-69.

Kato M, So I, Hishinuma Y, Nakamura O, Minami T (1992) Description and Synthesis of Facial Expression based on Isodensity Maps, in: Tosiyasu L (ed): Visual Computing, Springer-Verlag, Tokyo, pp.39-56.

Kurihara T, Arai K (1991), A Transformation Method for Modeling and Animation of the Human Face from Photographs, Proc. Computer Animation '91 Geneva, Switzerland, Springer-Verlag, Tokyo, pp. 45-57.

Lewis JP, Parke FI (1987), Automated Lipsync and Speech Synthesis for Character Animation, Proc. CHI '87 and Graphics Interface '87, Toronto, pp. 143-147.

Lewis J (1991) Automated Lip-sync: Background and Techniques, Journal of Visualization and Computer Animation, Vol.2, No4, pp.117-122.

Magnenat-Thalmann N, Primeau E, Thalmann D (1988), Abstract Muscle Action Procedures for Human Face Animation, The Visual Computer, Vol. 3, No. 5, pp. 290-297.

Magnenat-Thalmann N, Thalmann D (1987), The Direction of Synthetic Actors in the film Rendez-vous à Montréal, IEEE Computer Graphics and Applications, Vol. 7, No. 12, pp. 9-19.

Magnenat-Thalmann N, Thalmann D (1991), Complex Models for Visualizing Synthetic Actors, IEEE Computer Graphics and Applications, Vol. 11, No. 6.

Mase K, Pentland A (1990) Automatic Lipreading by Computer, Trans. Inst. Elec. Info. and Comm. Eng.,vol. J73-D-II, No. 6, pp. 796-803.

Nahas M, Huitric H, Saintourens M (1988), Animation of a B-Spline Figure, The Visual Computer, Vol. 3, No. 5, pp. 272-276.

Parke FI (1975), A Model for Human Faces that allows Speech Synchronized Animation, Computer and Graphics, Pregamon Press, Vol. 1, No. 1, pp. 1-4.

Parke FI (1982), Parametrized Models for Facial Animation, IEEE Computer Graphics and Applications, Vol. 2, No. 9, pp. 61-68.

Parke FI (1991), Control Parameterization for Facial Animation, Proc. Computer Animation '91, Geneva, Switzerland, Springer-Verlag, Tokyo, pp. 3-13.

Platt S, Badler N (1981), Animating Facial Expressions, Proc SIGGRAPH '81, pp. 245-252.

Saji H, Hioki H, Shinagawa Y, Yoshida K, Kunii TL (1992) Extraction of 3D Shapes from the Moving Human face Using Lighting Switch Photometry, in Magnenat Thalmann N , Thalmann D (eds) Creating and Animating the Virtual World, Springer-Verlag Tokyo, pp. 69-86.

Terzopoulos D, Waters K (1990) Physically Based Facial Modeling, Analysis, and Animation, Journal of Visualization and Computer Animation, Vol. 1, No. 2, pp. 73-80.

Terzopoulos and Waters (1991) Techniques for Realistic Facial Modeling and Animation, Proc. Computer Animation '91, Geneva, Switzerland, Springer-Verlag, Tokyo, pp. 59-74.

Turner R, Gobbetti E, Balaguer F, Mangili A, Thalmann D, Magnenat-Thalmann N, An Object-Oriented Methodology Using Dynamic Variables for Animation and Scientific Visualization, CG International '90, Springer Verlag, pp. 317-328.

Waters K (1987), A Muscle Model for Animating Three Dimensional Facial Expression, Proc SIGGRAPH '87, Vol. 21, No. 4, pp. 17-24.

Waters K, Terzopoulos D (1991) Modelling and Animating Faces using Scanned Data, Journal of Visualization and Computer Animation, Vol. 2, No. 4, pp. 123-128.
Williams L (1990), Performance Driven Facial Animation, Proc SIGGRAPH '90, pp. 235-242.